Provable Secure Generalized Signcryption

Xu an Wang[1], Xiaoyuan Yang[1] and Jindan Zhang[2]
[1] Key Laboratory of Information and Network Security,
Engneering College of Chinese Armed Police Force, 710086, P. R. China
[2] Department of Electronic Information,
Xianyang Vocational Technical College, 712000, P. R. China
E-mail:wangxahq@yahoo.com.cn

***Abstract*— Generalized signcryption which proposed by Han is a new cryptographic primitive which can work as an encryption scheme, a signature scheme or a signcryption scheme [5]. However,the security proof in their paper is uncorrect.our contribution are as following:First we give security notions for this new primitive.Secnond,we give an attack to [4]which is the first vision of [5] and propose an improved generalized signcryption scheme. Third, we give correct proofs for this new scheme.**

***Index Terms* – generalized signcryption, provable security, attack, security notions.**

I. INTRODUCTION

Along with developments of information society, security requirements for applications are usually both confidentiality and authentication. And these requirements have given birth of new research fields in cryptography, that is, how to combine confidentiality and authentication properly. A lot of work has been done in this field, such as how to encrypt message by block cipher properly to achieve authentication or how to combine ciphertext with signature properly to achieve authentication [1], [8]. Totally we can divide the work into three types: Encryption then Sign, Sign then Encryption, Encryption and Sign. In 1997, Zheng proposed a new cryptographic primitive: Signcryption [2]. The idea is compressing two independent operations (encryption and signature) in one operation (signcryption). There are three advantages from this transformation: reducing the steps needed by encryption and signature(less computation complexity); reducing length of ciphertext produced by encryption and signature(less communication complexity); reducing two modules of encryption and signature to one module of signcryption(less implementation complexity). Since then, a lot of research results have come out. We can see SCS-DSA, SCS-KCDSA signcryption scheme based on Discrete Logarithm problem, RSA-TBOS signcryption scheme based on Integer Factoring [6], ECSCS signcryption scheme based on elliptic curve [7], identity based signcryption scheme based on pairings.

In 2006, Han proposed a new primitive generalized signcryption [3]. The idea of this new primitive is still reducing, but this time, what's reducing is not the computation complexity or communication complexity, but the implementation complexity. Imagine this scenario, two users want to communicate safely. Sometimes they need both confidentiality and authentication, sometimes they just need confidentiality, and sometimes they just need authentication. If we adopt signcryption in this scenario, we must preserve module of encryption and module of signature for solely needing confidentiality or authentication. If we do not care very much about speed, we gain no remarkable advantage for adopting signcryption. Furthermore, adding something new to an established system seems no easy. But if we can embed encryption and signature in the signcryption module, we can easily encrypt or sign or signcrypt by only one module.

A. Motivation

Generalized Signcryption is the one which fits this goal. Generalized Signcryption is a new primitive which can work as an encryption scheme, a signature scheme, or a signcryption scheme. Maybe this can broaden the application range of signcryption.We must point out here that Generalized Signcryption can not substitute of encryption or signature. But it fit some particular application perfectly.

B. Related Works

Actually, the generalized signcryption concept is not new, it has been mentioned in Zheng's original paper [2]. In [20] Boyen et al proposed a mulitpurpose signcryption which they called as a swiss armed knife,the motivation is similar to our's. In [10], [11],Dodis et al proposed a versatile padding schemes which can perfectly played as an encryption or signature or signcryption scheme,The technique in their paper is padding message before processing. In the two extremities, the scheme turns to be OAEP-padding and PSS-padding. In the non-extremity, the scheme turns to be signcryption,furthermore, they prove their result is optimal, but they do not propose the generalized signcryption concept which is [5] main contribution.

C. Our contribution

However, [5] do not give the formal model for this new primitive and unfortunately the security proof for their scheme is uncorrect.Actually, all the papers [10], [11], [20]mentioned above do not consider formal security model for this multi-functionality cryptographic primitive. In this paper, we reconsider this new primitve thoroughly. our contribution are as following:First we give security notions for this new primitive.Secnond,we give an attack to [4]which is the first vision of [5] and propose an improved generalized signcryption scheme. Third, we give correct proofs for this new scheme.

doi:10.4304/jcp.5.5.807-814

D. Organization

The paper is organized as following: In the second section, we give new formal model for this new primitive which is based on the theory of provable security [14]–[19]. In the third section, we give an attack to the origin scheme in [4],which is the first vision of [5],and we give an improved scheme by give little change to the original scheme.In the forth section,we give formal correct proofs for this improved Generalized Signcryption scheme, which implies scheme in [5] be secure.We give our conclusion in the last section.

II. Generalized Signcryption and Its Security Notions

A. Definition of Generalized Signcryption and a Concrete Scheme ECGSC

Generalized Signcryption is a signcryption with more flexibility and practicability. It provides double Functions when confidentiality and authenticity are required simultaneously, and provides single Encryption or signature function when confidentiality Or authenticity is required only without any amended and additional computation. Namely, a generalized signcryption scheme will be equivalent to a signature scheme or an encryption scheme in special cases. Hence, a generalized signcryption will work in modes: signcryption, signature-only, and encryption-only.

Definition 1: Given a normal secure signature scheme $SIG = (Gen, Sig, Ver)$ where Gen is a key generation algorithm, $\tau \leftarrow Sig(m, SDK_S)$,$(T, \perp) \leftarrow Ver(\tau, VEK_S)$, a normal secure encryption scheme $ENC = (Gen, Enc, Dec)$ where Gen is the same algorithm as SIG's Gen,$\varepsilon \leftarrow Enc(m, VEK_R)$,$m \cup \{\perp\} \leftarrow Dec(\varepsilon, SDK_R)$ and a normal secure signcryption scheme $SC = (Gen, Sc, Usc)$ where Gen is the same algorithm as SIG's Gen,$w \leftarrow Sc(m, SDK_S, VEK_R)$,$(m \cup \{\perp\}) \cup (T, \perp) \leftarrow Usc(w, SDK_R, VEK_S)$.A generalized signcryption scheme $GSC = (Gen, Gsc, Ugsc)$ should be constructed satisfying the following:

1) KeyGen: Must be the same algorithm as Gen.
2) Generalized Signcryption: For $m \in M$,$w \leftarrow Gsc(m, SDK_S, VEK_R)$. When S is a special value, Gsc(m, SDK_S,VEK_R)=Enc(m, VEK_R); When R is a special value,Gsc(m,SDK_S,VEK_R)=Sig(m,SDK_S); When S and R are both not special values, Gsc(m, SDK_S, VEK_R)=Sc(m, SDK_S,VEK_R);
3) Generalized Unsigncryption: For $w \in \mathcal{C}$,$(m \cup \{\perp\}) \cup (T, \perp) \leftarrow Ugsc(w, SDK_R, VEK_S)$. When S is a special value,$Ugsc(w, SDK_R, VEK_S) = Dec(\varepsilon, SDK_R)$; When R is a special value,$Ugsc(m, SDK_S, VEK_R) = Ver(\tau, VEK_S)$; When S and R are both not special values, $Ugsc(w, SDK_R, VEK_S)$=$Usc(w, SDK_R, VEK_S)$.

Han proposed a Generalized Signcryption ECGSC based on ECDSA [4].Following is the scheme:

1) **Parameters**: Parameters of the elliptic curve
   - the parameters follow the SEC1 standard, which can be described as a sixtuple $T = (p, a, b, G, n, h)$;
   - G is a base point;
   - $ord(G) = n$;
   - O is the infinite element of group (G).
2) **Syntax**:In the scheme there are the syntax as following
   - $Q = [x]G$ denotes the scalar multiplex on the elliptic curve;
   - $\|$ denotes connecting two messages;
   - $\in_R$ denotes randomly choosing an element in one set;
   - $Bind$ denotes Alice and Bob's identity;
   - $\{0,1\}^l$denotes binary sequence of length l;
   - $K_{enc}, K_{mac}, K_{sig}$ is a binary sequence;
   - $H : \{0,1\}^* \rightarrow \mathbb{Z}_p^*$ and $K : \mathbb{Z}_p^* \rightarrow \{0,1\}^{\mathbb{Z}+*}$ denote two hash functions;
   - $LH(.) : \{0,1\}* \rightarrow \{0,1\}^{l+z}$ denotes hash function output long digest, we can choose $SHA-256$,$SHA-384$ or $SHA-512$;
   - $MAC_k : \{0,1\}^l \times \{0,1\}^t \times \{0,1\}^z$ denote message authenticate function which has key k. $|k| = t$, $|m| = l$,$l + |MAC(.)| = |LH(x_2)|$;
   - These hash functions have property :$H(0) \rightarrow 0$,$K(0) \rightarrow 0$,$LH(0) \rightarrow 0$,$MAC(0) \rightarrow 0$.
3) **Key generation**(n, T):Generate user's private and public key
   - Generate Alice's private and public key,choose $d_A \in_R \{1, \cdots, , n-1\}$,$Q_A = [d_A]G$,return (d_A, Q_A);
   - Generate Bob's private and public key,$d_B \in_R \{1, , n-1\}$,$Q_B = [d_B]G$,return (d_B, Q_A);
   - Generate null user's private and public key $(0, O) \leftarrow Gen(U, T)$,$U \in \Phi$.
4) **Generalized Signcryption** $SC(m, d_A, Q_B)$: it consists of seven algorithms
   - $k \in_R 1, \cdots, n-1$;
   - $R = [k]G = (x_1, y_1)$,$r = x_1 \bmod p$;
   - $[k]P_B = (x_2, y_2)$;
   - $K_{enc} = LH(x_2)$,$(K_{mac}, K_{sig}) = K(y_2)$;
   - If $d_A = 0$, $s = \phi$, Else $s = k_{-1}(H(m \parallel Bind \parallel K_{sig}) + rd_A) mod n$;
   - $e = MAC_{K_{mac}}(m)$;
   - $c = (m \parallel e) \oplus K_{enc}$,Return $w = (c, R, s)$.
5) **Generalized Unsigncryption** $DSC(w, d_B, Q_A)$: it also consists of seven algorithms
   - $r = x(R)$(R's x axiom);
   - $(x_2, y_2) = [d_B]R$;
   - $K_{enc} = LH(x_2), (K_{mac}, K_{sig}) = K(y_2)$;
   - $(m||e) = c \oplus K_{enc}$;
   - $e' = MAC_{K_{mac}}(m)$, If $e \neq e'$,return $\perp$else if $s = \phi$,return m;
   - $u_1 = s^{-1}H(m||Bind||K_{sig})$,$u_2 = s^{-1}r$;
   - $R' = [u_1]G + [u_2]Q_A$;If $R' \neq R$, return $\perp$,else return m.

B. Security Notions for Generalized Signcryption

Because Generalized Signcryption can work as encryption, signature or signcryption schemes, the adversary can get more oracles' service. For example, when considering confidentiality of Generalized Signcryption in encryption-mode, we must note adversary can get both Decryption Oracle service and Unsigncryption Oracle service. Note that Unsigncryption Oracle can maybe help the adversary decrypt challenge ciphertext. Analogously, when considering unforgeability of Generalized Signcryption in signature-mode, we must note adversary can get Signature Oracle service and Signcryption Oracle service. When considering confidentiality of Generalized Signcryption in signcryption-mode, we must note that the adversary can get Unsigncryption Oracle service and Decryption Oracle service. When considering unforgeability of Generalized Signcryption in signcryption-mode, we must note adversary can get Signature Oracle service and Signcryption Oracle service.

When talking about attacking against encryption schemes, we always emphasis on Decryption Oracle, but in fact, there is also an Encryption Oracle. But because public key is known to all, every one can get this Oracle's service, and it does not give the adversary any more attacking power than usual user. So we often omit this Oracle. The same thing happens in signature and signcryption schemes. Actually for Generalized Signcryption scheme, the adversary can get six types of Oracle's services: Encryption Oracle, Decryption Oracle, Signature Oracle, Verifying Oracle, Signcryption Oracle and Usigncryption Oracle.

Definition 2: **(Confidentiality in Encryption-mode)** Given security parameter $k = |p|$,let

$Adv^{IND-CCA2}_{GSC^{ENC},A}(k)$=$Pr[Exp^{IND-CCA2-1}_{GSC^{ENC},A}(k) = 1]$ - $Pr[Exp^{IND-CCA2-0}_{GSC^{ENC},A}(k) = 1]$

For $b \in \{0,1\}$,the following is the experiment:

Experiment $Exp^{ind-cca2-b}_{GSC^{ENC},A}(k)$
$pk_A, sk_A \leftarrow_R Gen(k, param)$;
$pk_B, sk_B \leftarrow_R Gen(k, param)$;
$(x_0, x_1, s) = A_1\{Enc_{pk_B}(.), Dec_{sk_B}(.), Sig_{sk_A}(.), Ver_{pk_A}(.), Gsc_{sk_A,pk_B(.)}, Ugsc_{sk_B,pk_A}(.)\}(find)$;
$y = GSC^{ENC}_{pk_B}(x_b)$;
$d = A_2\{Enc_{pk_B}(.), Dec_{sk_B}(.), Sign_{sk_A}(.), Ver_{pk_A}(.), Gsc_{sk_A,pk_B}(.), Ugsc_{sk_B,pk_A}(.)\}(x_0, x_1, y, s, guess)$;
Return d.

In the above attacking, A can get six services, the only restriction is that y cannot be queried to the Decryption Oracle $Dec_{sk_B}(.)$. If $Adv^{IND-CCA2}_{GSC^{ENC},A}(k)$is negligible, we say this Generalized Signcryption scheme is confidential when it work in encryption-mode.

Definition 3: **(Unforgeability in Signature-mode)** Given security parameter $k = |p|$, following is the experiment:

Experiment $ForgeExp^{cma}_{GSC^{SIG},F}(k)$
$pk_A, sk_A \leftarrow_R Gen(k, param)$;
$pk_B, sk_B \leftarrow_R Gen(k, param)$;
if$F^{Gsc_{sk_A,pk_B}(.),Ugsc_{sk_B,pk_A}(.)}_{Enc_{pk_B}(.),Dec_{sk_B}(.),Sig_{sk_A}(.),Ver_{pk_A}(.)}(.)$ output (m, s) which satisfy

- $Ver_{pk_A}(s) = T$;
- m has never been queried to $Sig_{sk_A}(.)$(existential unforgeable) or m is allowed to query $Sig_{sk_A}(.)$ but was never returned by $Sig_{sk_A}(.)$(strong unforgeable) ;

then return 1,else return 0.

In the above attacking, A can get six services, the only restriction is m has never been queried $Sig_{sk_A}(.)$(existential unforgeable) ,or m is allowed to query to $Sig_{sk_A}(.)$ but s was never returned by $Sig_{sk_A}(.)$(strong unforgeable). Let$Succ^{cma}_{Gsc^{SIG},F}(k) = Pr[Exp^{cma}_{GSC^{SIG},F}(k) = 1]$. If this value is negligible, we say this Generalized Signcryption scheme is unforgeable when it works in signature-mode.

Definition 4: **(Confidentially in Signcryption-mode)** Given security parameter $k = |p|$,let

$Adv^{IND-CCA2}_{GSC^{SC},A}(k)$=$Pr[Exp^{IND-CCA2-1}_{GSC^{SC},A}(k) = 1]$ -$Pr[Exp^{IND-CCA2-0}_{GSC^{SC},A}(k) = 1]$

For $b \in \{0,1\}$,the following is the experiment:

Experiment $Exp^{ind-cca2-b}_{GSC^{SC},A}(k)$
$pk_A, sk_A \leftarrow_R Gen(k, param)$;
$pk_B, sk_B \leftarrow_R Gen(k, param)$;
$(x_0, x_1, s) = A_1\{Enc_{pk_B}(.), Dec_{sk_B}(.), Sig_{sk_A}(.), Ver_{pk_A}(.), Gsc_{sk_A,pk_B(.)}, Ugsc_{sk_B,pk_A}(.)\}(find)$;
$c = GSC^{SC}_{pk_B,sk_A}(x_b)$;
$d = A_2\{Enc_{pk_B}(.), Dec_{sk_B}(.), Sign_{sk_A}(.), Ver_{pk_A}(.), Gsc_{sk_A,pk_B}(.), Ugsc_{sk_B,pk_A}(.)\}(x_0, x_1, c, s, guess)$;
Return d.

In the above attacking, A can get six services, the only restriction is that c was never queried $Ugsc_{sk_B,pk_A}(.)$.If $Adv^{IND-CCA2}_{GSC^{SC},A}(k)$ is negligible, we say this Generalized Signcryption scheme is confidential when it works in signcryption mode.

Remark 1 What's the diffirence between Definition 2 and Definition 4? In definition 2,the challenge ciphertext cannot be queried to *Decryption Oracle*, but we can transform challenge ciphertext into some valid signcryption ciphertext and then query it to the *Unsigncryption Oracle*. In definiton 4, the challenge signcryption ciphertext cannot be queried to *Unsigncryption Oracle*,but we can transform the challenge signcryption ciphertext to some valid ciphertext and then query it to the *Decryption Oracle*.

Definition 5: **(Unforgeablity in Signcryption-mode)** Given security parameter $k = |p|$, following is the experiment:
Experiment $ForgeExp^{cma}_{GSC^{SC},F}(k)$
$pk_A, sk_A \leftarrow_R Gen(k, param)$; $pk_B, sk_B \leftarrow_R Gen(k, param)$;
if $F^{Gsc_{sk_A,pk_B}(.),Ugsc_{sk_B,pk_A}(.)}_{Enc_{pk_B}(.),Dec_{sk_B}(.),Sig_{sk_A}(.),Ver_{pk_A}(.)}(.)$ output (m, C) which satisfy
$-$ m has never been queried to $Gsc_{sk_A,pk_B}(.)$;

$- Ugsc_{sk_B,pk_A}(C) = m$;
then return 1,else return 0.
In the above attacking, A can get six services, the only restriction is that c was never returned by $Gsc_{sk_A,pk_B}(.)$. Let$Succ^{cma}_{GSC^{SC},F}(k) = Pr[Exp^{cma}_{GSC^{SC},F}(k) = 1]$. If this value is negligible, we say the Generalized Signcryption scheme is unforgeable when it works in signcryption-mode.

Remark 2 What's the diffirence between Definition 3 and Definition 5? In definition 3,the forged signature is not the output of *signature Oracle*,but can be the transformation of some valid result returned by *Signcryption Oracle*. In definiton 5, the forged signcryption ciphertext is not the output of *Signcryption Oraxle* but can be the transformation of some valid result returned by *Signature Oracle*.

III. An Improved Generalized Signcryption Based on ECDSA

A. An attack on this Scheme and Some Remarks

Attack In the ECGSC scheme the adversary intercept the ciphertext $w = (c, R, s)$set $s = \phi$, query the new ciphertext $w = (c, R, \phi)$ to *Decryption Oracle*, the *Decryption Oracle* will return m, which break the confidentiality of Generalized Signcryption in signcryption-mode. Note here, the adversary does not query $w = (c, R, s)$ to *Unsigncryption Oracle*, which is the only restriction for the adversary. The attack can be successful just because we use *Decryption Oracle* to decrypt the modified challenge signcryption ciphertext.

Remark 3 The origin scheme depend on hash function with additional property, that is, $H(0) \to 0$,$K(0) \to 0$,$LH(0) \to 0$,$MAC(0) \to 0$.But we know, if there exists non-change point in hash function, this would bring bad effects to the hash function. Especially, for hash function working in CBC mode, this can be damage. Another reason is that hash function with addition property can not be easily devised. It does not follow principal of modern hash family. So we suggest deleting this additional property.

Remark 4 The original scheme uses if/else clause, and the conditional variant is s ,and s is just a local variant, programs with normal access rights can modify it. For example, some adversary can just add some program in the origin scheme's code at proper time, let $s = \phi$, he would get the plaintext m. So we suggest delete the if-clause in the algorithm.

B. An Improved Generalized Signcryption Based on ECDSA

In this section, we give an improved Generalized Signcryption scheme. Improved scheme has the same parameter, syntax with the origin scheme. But we do not need hash function satisfy $H(0) \to 0$,$K(0) \to 0$,$LH(0) \to 0$, $MAC(0) \to 0$, and we introduce another point Q, which can be any point not belonging to the elliptic curve (or no one would choose this point as his public key).Here we can assume $Q = (0, 0)$. The reason we introduce this point is for encryption-mode and signature-mode. We define a function $f(t)$. if $t = Q$, $f(t) = 0$,if $t \neq Q$, then $f(t) = 1$. For signcryption-mode, $Bind = SH(Q_A||Q_B)$, for encryption-mode, $Bind = SH(Q_A||Q)$,for signature-mode, $Bind = SH(Q||Q_B)$.SH represents hash function, its output is 32 bit, and we denote its length by $|sh|$. We change the length of LH's output to $l + z + |sh|$, we denote $|K_{sig}| = |sig|$.

1) **Parameters**: Same as the original scheme.
2) **Syntax**:Almost same as the original scheme except we do not need hash functions with additional property,introduce a new point and modify some syntex's meaning.
   - we do not need hash function satisfy $H(0) \to 0$,$K(0) \to 0$,$LH(0) \to 0$, $MAC(0) \to 0$;
   - we introduce another point Q, which can be any point not belonging to the elliptic curve (or no one would choose this point as his public key).Here we can assume $Q = (0, 0)$. The reason we introduce this point is for benefitting encryption-mode and signature-mode. We define a function $f(t)$. if $t = Q$, $f(t) = 0$,if $t \neq Q$, then $f(t) = 1$;
   - SH represents hash function, its output is 32 bit, and we denote its length by $|sh|$. We change the length of LH's output to $l + z + |sh|$, we denote $|K_{sig}| = |sig|$;
   - For signcryption-mode, $Bind = SH(Q_A||Q_B)$, for encryption-mode, $Bind = SH(Q_A||Q)$,for signature-mode, $Bind = SH(Q||Q_B)$.
3) **Key generation**(n, T):Same as the original scheme.
4) **Generalized Signcryption** $SC(m, d_A, Q_A, Q_B)$: it consists of seven algorithms
   - Compute $f(Q_A)$,$f(Q_B)$,
   - $k \in_R 1, \cdots, n-1$;
   - $R = [k]G = (x_1, y_1)$,$r = x_1 \bmod p$;
   - $[k]P_B = (x_2, y_2)$;
   - $K_{enc} = f(Q_B) * LH(x_2)$,$(K_{mac}, K_{sig}) = f(Q_B) * K(y_2)$;
   - If $d_A = 0$, $s = \phi$, Else $s = k_{-1}(f(Q_A) * H(m \parallel Bind \parallel K_{sig}) + f(Q_A) * rd_A) \ modn$;
   - $e = f(Q_B) * MAC_{K_{mac}}(m)$;
   - $c = (m \parallel e) \oplus K_{enc}$,Return $w = (c, R, s)$.
5) **Generalized Unsigncryption** $DSC(w, d_B, Q_A, Q_B)$: it also consists of seven algorithms
   - Compute $f(Q_A)$,$f(Q_B)$,
   - $r = x(R)$(R's x axiom);
   - $(x_2, y_2) = [d_B]R$;
   - $K_{enc} = f(Q_B) * LLH(x_2), (K_{mac}, K_{sig}) = f(Q_B) * LK(y_2)$;
   - $(m||e) = c \oplus K_{enc}$;
   - $e' = f(Q_B) * LMAC_{K_{mac}}(m)$, If $e \neq e'$,return $\perp$else if $s = \phi$,return m;
   - $u_1 = s^{-1} * f(Q_A) * H(m||Bind||K_{sig})$,$u_2 = s^{-1} * f(Q_A) * r$;
   - $R' = [u_1]G + [u_2]Q_A$;If $R' \neq R$, return $\perp$,else return m.

IV. Security Proofs for Our Improved Generalized Signcryption

The idea of the origin scheme's security proofs is the following. When the Generalized Signcryption work as in signcryption-mode, the author can reduce confidentiality of signcryption to a scheme proposed by Krawczyk in Crypto 2001 [1], and this scheme is proved to be ciphetext unforgeable under chosen plaintext attacks. We denote this encryption scheme ATEOTP and the analog Elliptic Curve's variant ECATEOTP. But the author just discussed the Signcryption Oracle service, no caring about other Oracle service, this is not sufficient. [5] can also reduce SUF-CMA of signcryption to SUF-CMA of ECDSA, but the reduction is uncorrect.Also [5] do not give secrurity proof for generalized signcryption working in encryption-mode and signature- mode.This paper tries to solve these problems.

A. Prove SUF-CMA of the Generalized Signcryption in Signcryption-mode

We will apply a standard technique of provable security theory game hopping in our proofs. We define a sequence of games:$G0$,$G1$. they are reduced from the real attacking game . In every game, the private and public key, the adversary and the Random Oracle's coin flipping space are not changed. The difference comes from the view defined by rules. We will reduce the attack to SUF-CMA of ECGSC to SUF-CMA of ECDSA. Assume the success probability of attacking SUF-CMA is τ, its running time is T. We denote character with $*$ as the forged ciphetext and its related variables.

GAME G0: In $GAMEG0$, we just use the standard technique of simulating hash function. We can know this environment and the really environment is indistinguishable in the random oracle model. Let S_0 denote attacking successfully, assume $Pr[S_0] = \varepsilon$.

1) Simulate Random Oracle $LH(x)$Query $LH(x)$,if the record (x, lh) is found in LH-list, then Oracle return lh else randomly choose $lh \in \{0,1\}^{l+z+|sh|}$,add (x, lh) to the H-list;
2) Simulate Random Oracle $K(y)$:Query $K(y)$,if the record (y, k) is found in K-list, then Oracle return k ,else randomly choose $k \in \{0,1\}^{z+|sig|}$,add (y, k) to K-list.
3) Simulate Random Oracle H:Query $H(m \parallel SH(Q_A \parallel Q_B) \parallel K_{sig})$,if the record $(m \parallel SH(Q_A \parallel Q_B) \parallel K_{sig}, h)$ is found in H-list, then Oracle return h ,else randomly choose $h \in \{0,1\}^{|p|}$ add record $(m \parallel SH(Q_A \parallel Q_B) \parallel K_{sig}, h)$ to H-list.
4) Simulate Random Oracle MAC:Query $MAC(K_{mac}, m \parallel SH(Q_A \parallel Q_B) \parallel s)$,If the record $(K_{mac}, m \parallel SH(Q_A \parallel Q_B) \parallel s, mac)$ is found in MAC-list, then Oracle return mac ,else randomly choose $mac \in \{0,1\}^z$,add the record $(K_{mac}, m \parallel SH(Q_A \parallel Q_B) \parallel s, mac)$ into the MAC-list.
5) Simulate Signcryption Oracle Sc:Real Signcryption in real environment. In assume adversary can get this service.
6) Simulate Unsigncryption Oracle Usc:Think about insider adversary. Because the adversary know the receiver's private key, he can get this integrated service (The simulator just gives the receiver's private key to the adversary).
7) Simulate Encryption Oracle Enc:Because the adversary can get the Encryption Oracle service by only needing to know the receiver's public key, but this is public to all. So the adversary can get the integrated service. (The simulator just gives the receiver's public key to the adversary).
8) Simulate Decryption Oracle Dec:Think about insider adversary. Because the insider adversary know the receiver's private key, he can get the integrated service. (The simulator just gives the receiver's private key to the adversary).
9) Simulate Sign And Verify Oracle Sig/Ver:In this game, assume the adversary can get the integrated service of Sign Oracle. Because implementing Verify Oracle just needs the signer's public key, and the public key is known to all. So the adversary can get this integrated service.
10) How to forge valid signcryption ciphertext:Assume the forged ciphetext is $w^* = (c^*, R^*, s^*)$the only restriction is that w^* was not queried to Sc Oracle.Totally there are two methods of forging ciphetext: One is by attacking signcryption directly, the other is utilizing Sign Oracle. Note the adversary can forge new valid signcryption ciphetext by utilizing Sign Oracle.

GAME G1: In this game, we will remove the restriction of linkage of encryption and signature in simulating GSC Signcryption Oracle. We remove the layer of encryption and reduce signcryption scheme to ECDSA signature scheme. We will substitute Sign Oracle by ECDSA algorithm. Other oracles are simulated as in $GAMEG0$.

1) Simulate Signcryption Oracle Gsc
   - Add new elements of $(\diamondsuit, (K_{mac}, K_{sig}))$ in K-list. Note we must set the first item of new element vacant; we give it some value later. Add new elements of $(\diamondsuit, K_{enc})$ in H-list. We also set the first item of new element vacant, we will give it some value later.
   - Call algorithm of $ECDSA(m \parallel SH(Q_A \parallel Q_B) \parallel K_{sig}, d_A)$in Random Oracle, let$(m \parallel SH(Q_A \parallel Q_B) \parallel K_{sig}, R, s)$be the output result. In this process there will be a H-list;
   - Find element of $(K_{mac}, m \parallel SH(Q_A \parallel Q_B) \parallel s)$in MAC-list. If $(K_{mac}, m \parallel SH(Q_A \parallel Q_B) \parallel s, K_{mac})$ is found in the MAC-list, then we return mac. Else, choosing randomly $mac \in \{0,1\}^z$return mac,add record of $(K_{mac}, m \parallel SH(Q_A \parallel Q_B) \parallel s, mac)$in MAC-list;
   - Compute $c = (m \parallel SH(Q_A \parallel Q_B) \parallel mac) \oplus K_{enc}$;

- Let (c, R, s) be the output of Signcryption Oracle Gsc when the input is (m, d_A, Q_A, Q_B);

2) Now we think about how to map vacant of elements in K-list and H-list to (x_2, y_2) . Because the simulator know the private key, so it can decryption the ciphertext. First we show how to simulate the Unsigncryption Oracle, in this process, we can give this map
3) Simulate Unsigncryption Oracle Ugsc
   - Query (c, R, s) to Unsigncryption Oracle Ugsc;
   - The simulator compute $(x_2, y_2) = d_B R$;
   - First we find s in the second item of $(K_{mac}, m \parallel SH(Q_A \parallel Q_B) \parallel s, mac)$ MAC-list. If s is found in $(K_{mac}, m \parallel SH(Q_A \parallel Q_B) \parallel s, mac)$, return $K_{mac} m \parallel SH(Q_A \parallel Q_B) \parallel s, mac$ else return "Invalid Ciphertext";
   - Next find K_{mac} in the second item of elements in K-list. If K_{mac} is found in $(\diamondsuit, (K_{mac}, K_{sig}))$-list, let the first item of this element be y_2, else return "Invalid Ciphertext";
   - Compute $t = c \oplus m \parallel SH(Q_A \parallel Q_B) \parallel mac$ and find t in the LH-list. If t is found equal to some element of $(\diamondsuit, K_{enc})$, then let the first item of this element be x_2, else return "Invalid Ciphertext".
4) Simulate Sign Oracle Sig:Using algorithm of $ECDSA(m \parallel SH(Q_A \parallel Q)), d_A)$,
   let its output be Sign Oracle's output.

Remark 5:In the above simulation,we use a technique different from usual. Here we use the condition that attacker can know the receiver's private key and can compute $[d_B]R$ and x_2, y_2.So we can find the relationship between x_2, y_2 and (K_{mac}, K_{sig}),K_{enc}.

$GameG1$ and $GameG0$ are indistinguishable, except some queries have been given to k-list,LH-list before simulation or some ciphertexts have been guessed correctly by adversary. Assume the adversary has queried K-Random Oracle,H-Random Oracle,LH-Random Oracle,MAC-Random Oracle $q_K, q_H, q_{LH}, q_{MAC}$
times, denote S_1 as the adversary forges successfully in GAME G1, then

$$| Pr[S_0] - Pr[S_1] | \leq \frac{q_H}{2^{|p|}} + \frac{q_{LH}}{2^{l+z+|SH|}} - \frac{q_H}{2^{|p|}} \cdot \frac{q_{LH}}{2^{l+z+|SH|}} \cdot \frac{q_{MAC}}{2^z} \cdot \frac{q_K}{2^{z+|Sig|}}$$

Theorem 1: If the adversary A can forge valid signcryption ciphertext of Generalized Signcryption in signcryption-mode successfully with probability τ and the running time is T.Assume A queries K-Random Oracle,H-Random Oracle, LH-Random Oracle, MAC-Random Oracle $q_K, q_H, q_{LH}, q_{MAC}$ times, queries Signcryption Oracle, Sign Oracle, Encryption Oracle, Unsigncryption Oracle, Verify Oracle, Decryption Oracle $q_{Gsc}, q_{Ugsc}, q_{Sig}, q_{Ver}, q_{Enc}, q_{Dec}$ times. Then he forges signature of ECDSA with probability ϵ,

$$\epsilon \geq \tau - (\frac{q_H}{2^{|p|}} + \frac{q_{LH}}{2^{l+z+|SH|}} - \frac{q_H}{2^{|p|}} \cdot \frac{q_{LH}}{2^{l+z+|SH|}} \cdot \frac{q_{MAC}}{2^z} \cdot \frac{q_K}{2^{z+|Sig|}})$$

The running time

$$T' \geq T + (q_{LH} + q_K) f + (q_{Gsc} + q_{Sig}) g$$

f denote the running time of compute$d_b R$ one time,g denote the running time of compute kG one time

B. Prove Confidentiality of the Generalized Signcryption in Signcryption-mode

We reduce confidentiality of the Generalized Signcryption in signcryption-mode to confidentiality of ECATEOTP which as following.

Definition 6: ECATEOTP is an encryption scheme, and we know it's IND-CCA2 secure [1].

1) Encryption $Enc(m, Q_A, Q_B)$
   - $k \in_R \{1, \cdots, n-1\}$;
   - $(x_1, y_1) = R = [k]G$
   - $(x_2, y_2) = [k]Q$;
   - $K_{enc} = LH(x2), (K_{mac}, K_{sig}) = K(y_2)$;
   - $e = MAC_{K_{mac}}(m \parallel SH(Q_A \parallel Q_B))$;
   - $c = (m \parallel SH(Q_A \parallel Q_B \parallel e) \oplus K_{enc}$;
   - Return $w = (c, R)$.
2) Decryption $Dec(w, d_B, Q_A, Q_B)$
   - $[d_B]R = (x_2, y_2)$;
   - $K_{enc} = LH(x2), (K_{mac}, K_{sig}) = K(y_2)$;
   - $(m \parallel SH(Q_A \parallel Q_B) \parallel e) = c \oplus K_{enc}$;
   - $e' = MAC_{K_{mac}}(m \parallel SH(Q_A \parallel Q_B))$;
   - if $e = e'$,return " $\perp$ ";else return m.

Assume the success probability of forging Valid Ciphertext of ECATEOTP is η , and running time is T.

GAME G0: In $GAMEG0$, we just use the standard technique of simulating hash function. We can know this environment and the really environment is indistinguishable in the random oracle model. Let S_0 denote attacking successfully, assume $Pr[S_0] = \gamma$.

1) Simulate Random Oracle $LH(x)$,$K(y)$,H,MAC: Same as common name oracles in section 4.1;
2) Simulate Signcryption Oracle Sc:Think about insider adversary. Because the adversary know the sender's private key, he can get this integrated service;
3) Simulate Unsigncryption Oracle Usc:Real Unsigncryption under real environment. Assume adversary can get this service;
4) Simulate Encryption Oracle Enc:The adversary can get the Encryption Oracle service by only needing to know the receiver's public key. And this is public to all, so the adversary can get this integrated service;
5) Simulate Decryption Oracle Dec:Assume the adversary can get this integrated service;
6) Simulate Sign And Verify Oracle Sig/Ver:Think about insider adversary. Because insider adversary know the

receiver's private key, he can get this integrated service.The adversary can get the Verify Oracle service by only needing to know the sender's public key, but this is public to all. So the adversary can get this integrated service.

7) How to decrypt challenge ciphertext:Denote the challenge ciphertext$(c*, R*, s*)$. There are two ways to decrypt the challenge ciphertext: One is to utilize attacking on the signcryption scheme. The other is to use Decryption Oracle.

GAME G1: In this game, we try to reduce Unsigncryption Oracle to Decryption Oracle of ECATEOTP and substitute Decryption Oracle of Generalized Signcryption by Decryption Oracle of ECATEOTP..

1) Simulate Signcryption Oracle Gsc
   - Everything is done honestly just as in the real Signcryption Algorithm. But when some queries to the Random Oracle LH, K, H, and MAC, we return something following the standard technique of simulating Hash Function.
2) Simulate Unsigncryption Oracle Ugsc
   - There have been LH, K, H, MAC-list in simulate Signcryption Oracle Gsc;
   - Using Decryption Oracle of ECATEOTP: $Dec(w, d_B, Q_A, Q_B)$in Random Oracle;
   - Algorithm Dec will compute $(x_2, y_2) = [d_B]R$,it must get value of $LH(x_2)K(y_2)$ according to LH-list, K-list. It finds (x_2, K_enc) and $(y_2, (K_{Mac}, K_{sig}))$ in K-list and LH-list. If the element is found, then return the second item of element; else return "Invalid Ciphertext";
   - Compute $(m \parallel Bind \parallel e) = c \oplus K_{enc}$;
   - Find $m \parallel SHQ_A \parallel Q_B \parallel K_{sig}$ in the first item of elements in H-List. If $(m \parallel SH(Q_A \parallel Q_B) \parallel K_{sig}, h)$ is found, Simulator return h.Else return "Invalid Ciphertext";.
   - Compute $u_1 = s^{-1} * hu_2 = s^{-1} * r$;
   - Compute $R' = [u_1]G + [u_2]Q_A$ If $R' \neq R$,return $\perp$else return m.
3) Simulate Decryption Oracle Dec:Using algorithm of $Dec(w, d_B, Q, Q_B)$, let its output be Decryption Oracle's output.

$GAMEG1$ and $GAMEG0$ are indistinguishable, except some ciphertexts have been guessed validly by adversary. Assume the adversary has queried K-Random Oracle, H-Random Oracle, LH-Random Oracle, MAC-Random Oracle $q_K, q_H, q_{LH},$
q_{MAC}times, denote S_1 as the adversary forges successfully in $GAMEG1$, then

$$|Pr[S_0] - Pr[S_1]| \leq \frac{q_H}{2^{|p|}} \cdot \frac{q_{LH}}{2^{l+z+|SH|}} \cdot \frac{q_{MAC}}{2^z} \cdot \frac{q_K}{2^{z+|Sig|}}$$

Theorem 2: If the adversary A can attack confidentiality of Generalized Signcryption in signcryption-mode successfully with probability η , the running time is T.Assume A queries K-Random Oracle, H-Random Oracle,LH-Random Oracle, MAC-Random Oracle times, queries Signcryption Oracle, Sign Oracle, Encryption Oracle, Unsigncryption Oracle, Verify Oracle, Decryption Oracle $q_{Gsc}, q_{Ugsc}, q_{Sig}, q_{Ver}, q_{Enc}, q_{Dec}$ times. Then he can attack IND-CCA2 property of ECATEOTP with probability

$$\zeta > \eta + \frac{q_H}{2^{|p|}} \cdot \frac{q_{LH}}{2^{l+z+|SH|}} \cdot \frac{q_{MAC}}{2^z} \cdot \frac{q_K}{2^{z+|Sig|}}$$

The running time

$$T' \geq T + (q_{LH} + q_K)f + (q_{Gsc} + q_{Sig} + q_{Ugsc}, q_{Ver}, q_{Enc}, q_{Dec})g$$

f denote the running time of computed_bR one time,g denote the running time of compute kG one time

C. Prove SUF-CMA of the Generalized Signcryption in Sgnature-mode

When Generalized Signcryption Oracle work as a signature scheme, Generalized Signcryption is actually ECDSA. So we omit the proof and give the following theorem.

Theorem 3: If the adversary A can attack SUF-CMA of Generalized Signcryption in signature-mode successfully with probability η, the running time is T. Then he can forge valid signature of ECDSA with probability

$$\mu \approx \eta$$

The running time $T' = T$.

D. Prove Confidentiality of the Generalized Signcryption in Encryption-mode

When Generalized Signcryption Oracle work as an encryption scheme, Generalized Signcryption is actually ECATEOTP. So we omit the proof and give the following theorem.

Theorem 4: If the adversary A can attack confidentiality of Generalized Signcryption in encryption-mode successfully with probability η, and the running time is T. Then he can forge valid ciphertext of ECATEOTP with probability

$$\mu \approx \eta$$

The running time $T' \approx T$.

V. Conclusion and Open Problems

Based on Han et al's paper [3]–[5] our paper pay attention to the formal model of Generalized Signcryption. We give an improved Generalized Signcryption scheme based on ECDSA and give its security proof . We remark that this paper just gives a Generalized Signcryption scheme based on ECC, there are still much work can be done on this new primitive.So we propose following open problems to develop generalized signcryption research.

1) Give more experiments on the efficiency advantage over solely signcryption.
2) Propose more generalized signcryption schemes based on discrete logarithm problem.

3) Propose generalized signcryption schemes based on integer factoring problem.
4) Propose generalized signcryption schemes based on identity-based cryptography([21] has partially solved this question,but we can hope more).
5) Consider universal compose security for generalized signcryption. And this maybe be quite complicated for this cryptographic primitive can not lie in the current framework of universal composable security.

ACKNOWLEDGMENT

This work is supported by the National Natural Science Foundation of China under contract no. 60842006. The authors would like to express their gratitude thanks for Dr. Yiliang Han's many helpful discussions.

REFERENCES

[1] H. Krawczyk The order of encryption and authentication for protecting communications (or: How secure is SSL?). In *Advances in Cryptology, Proc. CRYPTO2001,* LNCS 2139, pages 310–331. Springer–Verlag, 2001.
[2] Y. Zheng Digital signcryption or how to achieve cost (signature &encryption) $\ll$ cost (signature) + cost (encryption). In *Advances in Cryptology, Proc. CRYPTO 1997,* LNCS 1294, pages 165–179. Springer–Verlag, 1997.
[3] Y. Han, X. Yang. ECGSC: Elliptic Curve based Generalized Signcryption Scheme,Cryptology Eprint Archive, 2006/126.
[4] Y. Han, X. Yang. New ECDSA-Verifiable Generalized Signcryption. *Chinese Journal of Computer, No. 11.*, pages. 2003–2012, 2006.
[5] Y. Han. Generalization of Signcryption for Resources-constrained Environments. *Wireless Communication and Mobile Computing*, pages. 919–931, 2007.
[6] J. Malone-Lee, Mao W. Two birds one stone: Signcryption using RSA. In *Topics in Cryptology - CT-RSA 2003,* LNCS 2612, pages. 210–224. Springer–Verlag, 2003.
[7] Y. Zheng, H. Imai. How to construct efficient signcryption schemes on elliptic curves. *Information Processing Letters,, Vol. 68, No. 5, Sep.*, pages. 227–233, 1998.
[8] M. Bellare, C. Namprempre. Authenticated encryption: relations among notions and analysis of the generic composition paradigm. In *Advances in Cryptology, Proc. ASIACRYPT 2000,* LNCS 1976, pages 531–545. Springer–Verlag, 2000.
[9] J.H. An, Y. Dodis and T. Rabin On the security of joint signature and encryption.In *Advances in Cryptology, Proc. EUROCRYPT 2002,* LNCS 2332, pages 83–107. Springer–Verlag, 2002.
[10] Y. Dodis, M. Rreedman, S. Jarecki and S. Walfish, Optimal signcryption from any trapdoor permutation. Cryptology ePrint Archive, Report: 2004/020, 2004.
[11] Y. Dodis, M. Rreedman, S. Jarecki, and S. Walfish, Versatile padding schemes for joint signature and encryption. In *Proceedings of Eleventh ACM Conference on Computer and Communication Security (CCS2004)*, pages 196–205. IEEE Computer Society, 2004.
[12] D. Alexander. Hybrid Signcryption Schemes With Outsider Security.In *Proceedings of The 8th Information Security Conference (ISC 2005),* LNCS 4212, pages. 203–217, Springer–Verlag, 2005.
[13] D. Alexander. Hybrid Signcryption Schemes With Insider Security. In *Proceedings of Information Security and Privacy 2005) (ACISP 2005),* LNCS 4307, pages. 253–266, Springer–Verlag, 2005.
[14] M. Bellare, P. Rogaway. Random oracle are practical: a paradigm for designing efficient protocols.In *Proceeding of the First ACM Conference on Computer and Communication Security (CCS1993)*, pages.62–73, IEEE Computer Society, 1993.
[15] J. Baek, R. Steinfeld and Y. Zheng. Formal Proofs for the Security of Signcryption.In *Public Key Cryptography'02 (PKC 2002),* LNCS 2274, pages. 80–98, Springer–Verlag, 2002.
[16] J. Stern, D. Pointcheval, J. Malone-Lee and N.Smart. Flaws in Applying Proof Methodologies to Signature Schemes.In *Advances in Cryptology-Crypto'02 (CRYPTO 2002),* LNCS 2442, pages. 93–110, Springer–Verlag, 2002.
[17] M. Bellare and P. Rogaway. Optimal Asymmetric Encryption - How to Encrypt with RSA.In *(Eurocrypt'94),* LNCS 950, pages. 92–111, Springer–Verlag, 1995.
[18] M. Bellare and P. Rogaway. The Exact Security of Digital Signatures -How to Sign with RSA and Rabin. In *(Eurocrypt '96),* LNCS 1070, pages. 399–416, Springer–Verlag, 1996.
[19] J. Baek, R. Steinfeld and Y. Zheng, Formal Proofs for the Security of Signcryption.*Journal of Cryptology, Vol. 20, Issue 2*, pages. 203–235, 2007.
[20] X. Boyen. Multipurpose identity-based signcryption:a swiss army knife for identity-based cryptography.In *(Crypto03)*, pages. 382–398, Springer–Verlag, 2003.
[21] L. Sunder and K. Prashant. ID based generalized signcryption,Cryptology Eprint Archive, 2008/084.

Xu an Wang was born in Feb. 23th, 1981. He obtained his bachelor and master's degree in the Engneering College of Chinese Armed Police Force. Now he is a lecturer in the same college, his main research fields are cryptography and information security.

Xiaoyuan Yang was born in Jan. 2th, 1959. He obtained his bachelor and master's degree in the Xidian University. Now he is a Professor in the Engneering College of Chinese Armed Police Force, he has pubilished almost 30 papers on different conferences and journals, his main research fields are cryptography and information security.

Jindan Zhang was born in April. 29th, 1983. She obtained her bachelor's degree in the Xidian University and her master's degree in the Shanxi University of Technology and Science. Now she is a lecturer in the Xianyang Vocational Technical College, her main research fields are digital watermark, cryptography and information security.